



**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

INVENTOR: Juei Chang et al.
CASE: P3925
SERIAL NO.: 09/629,492 **GROUP ART UNIT:** 2178
FILED: 07/31/2000 **EXAMINER:** Campbell, Joshua D
SUBJECT: Software Bundle for Providing Automated Functionality to a WEB-Browser

PARTY IN INTEREST: All inventions in the disclosure in the present case are assigned to or assignable to: Yodlee.com, Inc.

Commissioner for Patents
Po Box 1450
Alexandria, Va 22313-1450

Dear Sir:

APPEAL BRIEF

11/13/2006 MBIZUNES 00000040 09629492

03 FC:2402

250.00 OP

1.0 Real Party in Interest

The real party in interest is Yodlee.com, Inc.

2.0 Related Appeals and Interferences

This is an appeal from the Office Action of the Examiner dated July 17, 2006, finally rejecting claims 1, 2, 4-7, 9-11 and 13-21, the only pending claims in the application. There are no related appeals of the claims in this case or interferences in the instant case.

3.0 Status of the Claims

Following is the status of all claims in the instant case:

1. Rejected - appealed in this brief; independent.
2. Rejected - appealed in this brief; dependent.
3. Canceled
4. Rejected - appealed in this brief; dependent.
5. Rejected - appealed in this brief; dependent.
6. Rejected - appealed in this brief; dependent.
7. Rejected - appealed in this brief; dependent.
8. Canceled
9. Rejected - appealed in this brief; dependent.
10. Rejected - appealed in this brief; independent.
11. Rejected - appealed in this brief; dependent.
12. Canceled
13. Rejected - appealed in this brief; dependent.
14. Rejected - appealed in this brief; dependent.

15. Rejected - appealed in this brief; dependent.
16. Rejected - appealed in this brief; dependent.
17. Rejected - appealed in this brief; independent.
18. Rejected - appealed in this brief; dependent.
19. Rejected - appealed in this brief; dependent.
20. Rejected - appealed in this brief; dependent.
21. Rejected - appealed in this brief; dependent.

4.0 Status of Amendments

No amendments have been filed subsequent to the rejection of claims 1, 2, 4-7, 9-11 and 13-21, the subjects of this appeal.

5.0 Summary of the Claimed Subject Matter

Following is a concise explanation of the subject matter defined in each of the four standing independent claims including their dependent claims.

5.1 Independent claim 1

1. A software-bundle (Fig. 1, 35; pg. 7, lines 12-13; fig. 13) residing on a server for navigating on a data network (Fig. 1, Internet 13) on behalf of a user by proxy, comprising:

a browser application (Fig. 13, Y-Browser, Fig. Browser 306) for navigating on the network;

a set of functional programs for performing tasks (Figs. 13-18, 307-317; Pg. 72 API Software bundle);

a set of APIs for integrating the functional programs to the browser application according to a machine-readable set of instructions (pg. 72, API Software bundle, lines

19-21); and

a control application for spawning, managing and terminating an instance of the browser application and monitoring behavior of the browser instance during a navigation sequence (Fig. 13, Robot controller 321, Browser 306; pg. 74, lines 7-14);

wherein the software-bundle functions as a navigation system capable of fully automating the functionality of a manual navigation system controlled by a user in such a way that allows the system to perform normal user navigation and registration actions automatically and transparent to the user including navigation to Web sites, registration with a user name and password, form filling, data searching, parsing Web pages and data capture and return, thereby reducing or eliminating the chance of failed user registration, and the set of machine-readable instructions is provided from an external source other than the control application (pg. 72, API Software bundle).

In summary, the above claim provides an automated navigation to by spawning, controlling and managing a browser application to gather information specific to a user by performing truly automatic navigation procedures including registration with a user name and password, form filling, data searching, parsing Web pages and data capture and return.

5.2 Independent method claim 10

10. A method for performing an automated navigation sequence on a data network comprising the steps of:

(a) providing a machine-readable set of instructions, by a software-control application for initiating, running, and closing the navigation sequence from a server on the data network (Fig. 13, pg. 74, lines 7-14; Figs. 14-18);

(b) executing an instance of a browser application, the execution resulting from receipt of the machine-readable set of instructions (pg. 72, API Software bundle, lines 19-21);

(c) executing and completing a series of tasks during the navigation sequence, including navigation, registration with a user name and password, form filling, data searching, parsing Web pages and data capture and return, according to the order of instruction contained in the machine-readable set of instructions (Figs. 14-18); and

(d) terminating the instance of browser application, the termination resulting from the completion of the machine-readable set of instructions by the instance of browser application (pg. 74, lines 7-14).

In summary, method claim 10 provides for a automatic navigation sequence specific to a user by controlling a browser application completing a series of automated tasks including navigation, registration with a user name and password, form filling, data searching, parsing Web pages and data capture and return and spawning and terminating the instance of the browser application.

5.3 Dependent claims 11, 13-21

Claims 11, 13-21 provide for (13) wherein in step (a), the machine readable instructions provide for monitoring the navigation sequence by the software-control application (Fig. 15, lines 25-28); (14) wherein in step (b), the machine-readable set of instructions contains a first instruction for spawning an instance of the browser application; (15) wherein in step (d), the machine-readable set of instructions contains a last instruction for closing an instance of the browser application; (17) wherein the API's further provide automated extensibility to the browser instance by accessing and utilizing any Java-based routines that are used during navigation on a network; (18) wherein the API's extend automated browser functions by including at least one of searching Web site destinations, emulating all user input actions, error recovery and statistic-collection of a navigation sequence (Fig. 18, lines 1-9); (19) wherein the functional programs include at least one of Web page data parsing, image search, failure-detection and dialog intercept 9Fig. 18, lines 1-19; Fig. 13, 311; pg. 73 lines 15-21); (20) wherein the search function

includes determining a data structure tree defining how data is displayed on a WEB page in HTML format (Fig. 16,pg. 77 lines 14-22); (21) The software bundle of claim 1 wherein registration occurs automatically by generating and submitting usernames and passwords according to form requirements and site rules at the Web sites, wherein the user has no knowledge of the data even after successful registration (pg. 57, lines 5-11).

6. Grounds of Rejection to be Reviewed on Appeal

In the Office Action mailed 07/17/2006, as presented by Examiner Campbell, Joshua D., claims 1-2, 4-7, 9-11 and 13-19, and 21 are finally rejected under 35 U.S.C. 103(a) as being unpatentable over Burson et al. (US Patent Number 6,405,245, filed on October 28, 1998) hereinafter Burson. Claim 20 remains rejected under 35 U.S.C. 103(a) as being unpatentable over Burson, and further in view of Thompson et al. (US Patent Number 6,571,253, filed on April 28, 2000) hereinafter Thompson.

7. Argument

Following is a presentation of arguments against the rejection put forth by the Examiner.

7.1 35 U.S.C. 103 against claims 1-2, 4-7, 9-11 and 13-20, and 21

The Examiner's Arguments:

Regarding claim 1 and 10:

(A) Burson et al. discloses a method in which processing components (functions) of a P1 engine are used to perform tasks automatically, the tasks include navigating to websites, registering with the websites with a username and password which includes form filling (form filling includes searching and parsing the data fields in the form) and providing the user with data from the finished tasks (column 3, lines 15-47, column 4, line 66-column 5, line 21 and column 7, lines 30-67 of Burson et al.).

(B) Burson et al. discloses a method in which the processing components are integrated into browser functionality (column 4, lines 36-65 of Burson et al.).

(C) The P1 engine generates a simulated web client (browser instance) to perform tasks which are monitored, and when the task is completed the data is returned to the user interface of the browser application at which point the simulated web client is terminated and control is returned to the user interface (column 7, lines 30-67 of Burson et al.).

(D) Burson et al. does not disclose the use of an API for integration purposes. However, it would have been obvious to one of ordinary skill in the art at the time the invention was made to use an API to allow an application such as a browser to operate in conjunction with separate processing components (i.e. Java applets — column 8, lines 13-45 of Burson et al.) because APIs are commonly used to provide communication between applets in Java virtual machine.

Regarding claim 13-16

Burson et al. discloses a method in which a P1 (personal information) engine (control application) will perform browser transactions invisible to the user (column 7, lines 30-67). The P1 engine generates a simulated web client (browser instance) to perform tasks which are monitored, and when the task is completed the data is returned to the user interface of the browser application at which point the simulated web client is terminated and control is returned to the user interface (column 7, lines 30-67 of Burson et al.).

Regarding dependent claims 17 and 18

Burson et al. discloses that automated browser functions include emulating all user input actions during navigation (column 10! lines 4-43 of Burson et al.) Burson et al. does not disclose the use of an API for integration purposes with the different functions. However, it would have been obvious to one of ordinary skill in the art at the time the invention was made to use an API to allow an application such as a browser to operate in conjunction with separate functions (i.e. Java applets - column 8, lines 13-45 of Burson et

al.) because APIs are commonly used to provide communication between applets in Java virtual machine.

Regarding dependent claim 19

Burson et al. discloses a method in which the functional programs intercept the dialog necessary to navigate (i.e. cookie information) (column 8, lines 4-65 of Burson et al.).

Regarding dependent claim 21

Burson et al. discloses a method in which the information used to fill in the form may be generated automatically (column 6, lines 41- 44 of Burson et al.). Burson et al. discloses a method in which a P1 (personal information) engine (control application) will perform browser transactions, which include navigation and registration transactions, invisible to the user (column 7, lines 30-67).

Regarding Claim 20

Claim 20 remains rejected under 35 U.S.C. 103(a) as being unpatentable over Burson in view of Thompson. Burson et al. does not disclose displaying the data structure in a tree format as part of the search function. However, Thompson et al. discloses a method in which in order to perform a search an HTML document is first broken down into a DOM tree which defines the hierarchal structure of the display of the document (column 2, lines 1-65 of Thompson et al.). It would have been obvious to one of ordinary skill in the art at the time the invention was made to have combined the method of searching of Thompson et al. with the application of searching of Burson et al. because it would have increased the expressive power for locating the data item of interest.

Appellant's response:

(A) Applicant argues that the browser and instances of the browser as claimed are

not controlled in Burson. In Burson, the PI engine performs all functions of the system internally and delivers the data to the user's browser interface. In appellant's application, as claimed, the external browser used for navigation by the user is the browser application controlled by the API's. A user does not navigate with any part of the PI engine of Burson. The browser function in Burson is limited to the user's premises (Burson, client software 270, col. 4, lines 43-57). Applicant argues that Burson fail to teach or suggest searching and parsing for data and/or images, as claimed because the information held in the PI store points directly to the desired information or a trial/error method is used (Burson col. 6, lines 35-65; col. 7 40-57).

(B) Applicant argues there is absolutely no disclosure in Burson, including portions relied upon by the Examiner, teaching or suggesting integrating APIs with the browser functionality. In Burson, as argued above, the PI engine performs all navigation. No integration between APIs and browser functionality occurs in Burson.

(C-D) Applicant argues that the present invention operates within a conventional browser application. Burson's Web client does not read on the browser application as disclosed and claimed in the present invention. The client software 270 and the Web client of Burson are not the same application. Burson fails to teach a set of APIs for integrating the functional programs to the browser application. Burson's Web client has all functionality integrated within, no integration is necessary. In applicant's invention, the API bundle can integrate with existing browsers which are not proprietary to the system, this is an important patentable distinction that Burson fails to teach. Because Burson fails to teach integrating a software bundle with a Web browser, as claimed, there is absolutely no motivation in Burson, or what is generally known in the art to use an API to allow an application such as a browser to operate in conjunction with separate processing components as espoused by the Examiner without the consideration of hindsight knowledge of applicant's disclosure.

Regarding claim 13-16

As argued above, because Burson fails to teach integration or using a browser

application, as known in the art, machine readable instructions provided to the software-control application manipulating the browser for monitoring the navigation sequence, spawning an instance of the browser application, and closing an instance of the browser application, as claimed are not taught or suggested in Burson.

Regarding dependent claims 17 and 18

The arguments provided above, regarding Burson and the absent teaching of integrating API function modules with a browser application, suffice to rebut this argument.

Regarding dependent claim 19

Applicant argues that the cookies as disclosed in Burson fail to teach or suggest Web page data parsing, image search, failure-detection and dialog intercept, as claimed. This is pure conjecture on the Examiner's part with absolutely no teaching in Burson to teach said limitation (Burson col. 5, lines 30-43).

Regarding claim 20

Applicant argues that because Burson fails to teach an ability to search, as claimed, the motivation for the combination with Thompson does not exist. Applicant asserts the prior art must teach all of applicant's claimed limitations, and the motivation to combine may be found in what's generally known in the art. In this case the Examiner has failed to provide any prior art capable of performing automated search of a user's personal information on a Web page by controlling a browser application.

Regarding claim 21

Applicant argues that Burson does not teach or suggest generating *any* input data, whatsoever, required to retrieve a user's personal data. In Burson all data needed must be held in the PI store. If data is needed and it is not in the PI store, the user is notified to update the information or to manually input the information (Burson col. 8, lines 13-32).

8.Claims Appendix

The claims involved in the appeal are:

1. A software-bundle residing on a server for navigating on a data network on behalf of a user by proxy, comprising:

a browser application for navigating on the network;

a set of functional programs for performing tasks;

a set of APIs for integrating the functional programs to the browser application according to a machine-readable set of instructions; and

a control application for spawning, managing and terminating an instance of the browser application and monitoring behavior of the browser instance during a navigation sequence;

wherein the software-bundle functions as a navigation system capable of fully automating the functionality of a manual navigation system controlled by a user in such a way that allows the system to perform normal user navigation and registration actions automatically and transparent to the user including navigation to Web sites, registration with a user name and password, form filling, data searching, parsing Web pages and data capture and return, thereby reducing or eliminating the chance of failed user registration, and the set of machine-readable instructions is provided from an external source other than the control application.

2. The software-bundle of claim 1 wherein the data network is the Internet network.

3. (Canceled)

4. The software-bundle of claim 1 wherein the set of machine-readable instructions is provided to the bundle by the control application.

5. The software-bundle of claim 4 wherein the set of machine-readable instructions

covers a single navigation sequence.

6. The software-bundle of claim 5 wherein the set of machine-readable instructions covers a series of navigation sequences.

7. The software-bundle of claim 6 wherein the bundle resides on a single processor and includes an instance of the control application.

8. (Canceled)

9. The software-bundle of claim 6 wherein the software-bundle shares a control application with other like software-bundles executing on other processors.

10. A method for performing an automated navigation sequence on a data network comprising the steps of:

(a) providing a machine-readable set of instructions, by a software-control application for initiating, running, and closing the navigation sequence from a server on the data network;

(b) executing an instance of a browser application, the execution resulting from receipt of the machine-readable set of instructions;

(c) executing and completing a series of tasks during the navigation sequence, including navigation, registration with a user name and password, form filling, data searching, parsing Web pages and data capture and return, according to the order of instruction contained in the machine-readable set of instructions; and

(d) terminating the instance of browser application, the termination resulting from the completion of the machine-readable set of instructions by the instance of browser application.

11. The method of claim 10 wherein the data network is the Internet network.

12. (Canceled)

13. The method of claim 10 wherein in step (a), the machine readable instructions provide for monitoring the navigation sequence by the software-control application.

14. The method of claim 13 wherein in step (b), the machine-readable set of instructions contains a first instruction for spawning an instance of the browser application.

15. The method of claim 13 wherein in step (d), the machine-readable set of instructions contains a last instruction for closing an instance of the browser application.

16. The method of claim 13 wherein in step (b), the browser instance is spawned by the software-control application, and in step (d), the browser instance is terminated by the software-control application.

17. The software bundle of claim 1 wherein the API's further provide automated extensibility to the browser instance by accessing and utilizing any Java-based routines that are used during navigation on a network.

18. The software bundle of claim 17 wherein the API's extend automated browser functions by including at least one of searching Web site destinations, emulating all user input actions, error recovery and statistic-collection of a navigation sequence.

19. The software bundle of claim 1 wherein the functional programs include at least one of Web page data parsing, image search, failure-detection and dialog intercept.

20. The software bundle of claim 18, wherein the search function includes determining a data structure tree defining how data is displayed on a WEB page in HTML format.

21. The software bundle of claim 1 wherein registration occurs automatically by generating and submitting usernames and passwords according to form requirements and site rules at the Web sites, wherein the user has no knowledge of the data even after successful registration.

9. Evidence Appendix

No evidence other than the arguments and facts presented in this brief is provided.

10. Related Proceedings Appendix

No copies provided, because these claims have never been appealed.

Respectfully Submitted,
Juei Chang et al.

By /Donald R. Boys/
Donald R. Boys
Reg. No. 35,074

Central Coast Patent Agency, Inc.
3 Hangar Way, Suite D
Watsonville, CA 95076
831-768-1755